Chapter 8. Building-Data Import Tool (BIT)

The building-data import tool (**BIT**) is a utility that is designed to help you import large databases of property information and to process that data so as to be able to create occupancy to model building type relationships. It can read a variety of different types of database formats and configurations and will translate these into a standard format for use by **HAZUS**. The **BIT** includes a utility that allows you to run queries on databases so that you can identify certain types of properties (e.g. unreinforced masonry) or gather information about buildings with certain characteristics.

8.1 Getting Your Data in the Right Format

Before running **BIT** you need to ensure that your data is in a form that the program can process. For example, if you have purchased tax assessor's files on magnetic tape, you will have to have those tapes read and transferred to floppy disk or CD-ROM. You will need to convert your database to a *.dbf format if it is in some other database format such as *.db, *.xls, etc. Many database management programs have the option of saving data in a *.dbf format, so this may be relatively simple. Another problem that can occur is that square foot building area is not reported as a single number but instead a sub-area is given for each floor or each portion of the building. In this case you will have to sum the individual sub-areas for each building and put the total building area in a single field. In the case of commercially available property data, you will need to extract the records from the database using software supplied by the vendor. Other problems you may encounter are appearance of properties more than once if they have multiple owners, or the reporting of multi-building complexes, and the use of two or three different occupancy definitions for a single property. All of these will require judgment on your part, and some of these problems will be very challenging.

BIT can only work with the following three types of files:

- ASCII delimited (*.asc)
- Dbase file (*.dbf)
- Fixed length file (*.txt)

If your database is not in one of these three formats, you will need to use an external database management program to convert your data into one of these formats.

The **BIT** can only import data from one county at a time. If your data file contains properties from multiple counties, you will need to use a database management program to sort the data by county and organize the data into separate files for each county.

8.2 Starting BIT

BIT can be launched in two ways: either from within **HAZUS** or stand-alone.

To launch **BIT** from within HAZUS, select the command **Inventory|General Building Stock|Building Import Tool (BIT).**

To launch **BIT** by itself, select **Start|FEMA Risk Assessment System|BIT.** This assumes that the default group (**FEMA Risk Assessment System)** was used during installation.

8.3 Specifying the Input File

After starting the **BIT**, you will be presented with the window shown in Figure 8.1. This window guides you through the five steps needed to develop the occupancy to model building type relationships for your region. The first step in the process is to specify the property data file you will be using. To start this step click on the **Specify Input File...** button.

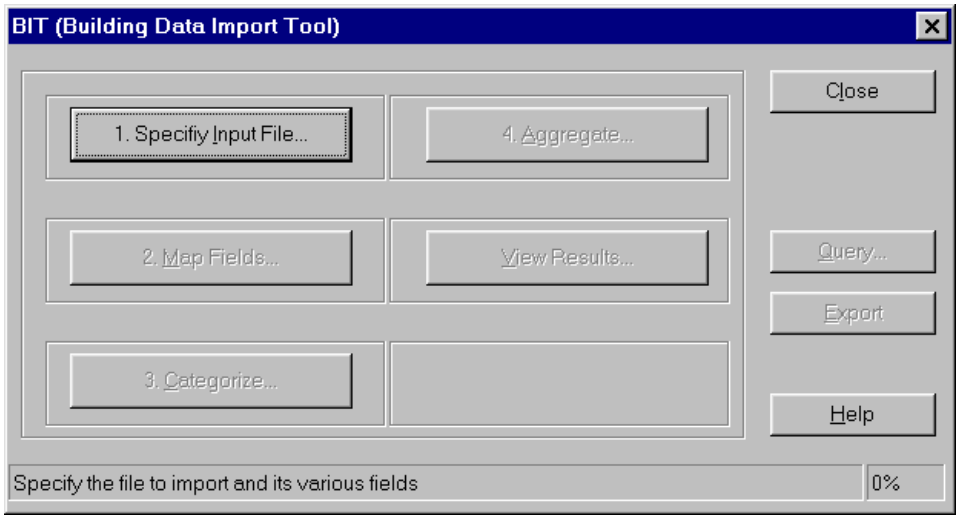


Figure 8.1 Building data import tool main menu

You will be asked to select an input file. You can choose from one of the four following options: ASCII delimited (*.asc), Dbase file (*.dbf), fixed length file (*.txt), and configuration file (*.bcf). A configuration file is generated by the **BIT**, and is available only if you have started the import process previously, but did not complete all five steps. The *.bcf file allows you to continue an incomplete import activity without starting over from the beginning.

8.3.1 Importing an ASCII Delimited Database

After you click on the **Specify Input File...** button in Figure 8.1, you will be presented with the window shown in Figure 8.3. Suppose that the particular property data file that you want to import is an ASCII delimited file. A delimited file is one that uses a specific character to separate the fields of information. Delimited files come with a variety of different characters to separate the fields. The most common are the comma and the tab. However, the delimiter can be any character. An example of two records from an ASCII delimited file is shown here:

```
"521-525 Main St","Anytown","94102-1102","121.00","Store
Building",4195,"1",2,"883263",16,"79","","880720","C","Concrete","Stucc
o","Concrete","Steel","Flat", "Built-up","","Average","$357",",","0284-
000"
```

```
"332 North St","Anytown","94102-
2607","125.00","Apartment",16030,"6",24,
"341314",23,"72","72","830404","C","Concrete","Concrete","Concrete","Co
ncrete",
"Flat","Tar & Gravel","","Fair","$17",",","0333-001"
```

Figure 8.2 Two records from an ASCII delimited file

Each record shown in Figure 8.2 spans four lines with each field separated by a comma. Quotes are used to indicate alphanumeric (text) data and entries without quotes are numbers. The **BIT** is capable of distinguishing these two types of inputs and it shouldn't cause you any problems when both types appear in the same record. It is important to understand that the **BIT** can recognize this file as ASCII delimited only if you specify the filename extension as .asc.

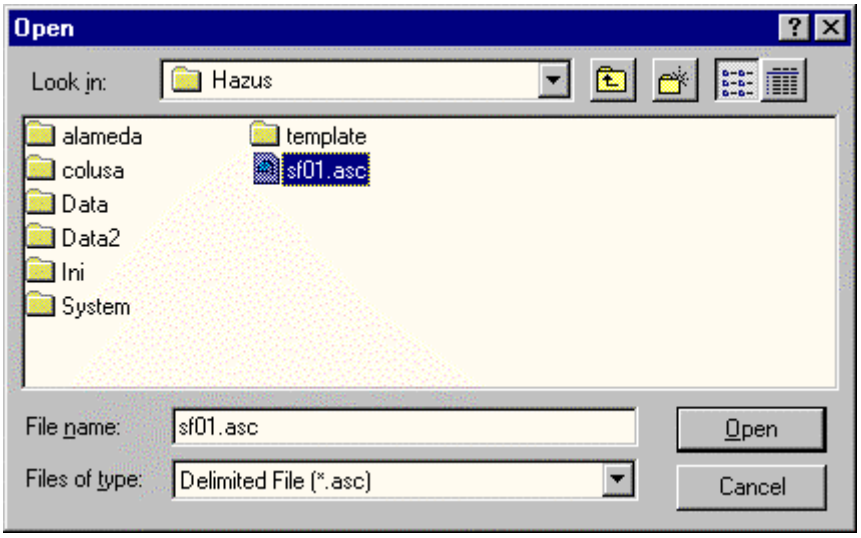


Figure 8.3 Specifying a *.asc input file in the building-data import (BIT) tool

After you have specified the file name in Figure 8.4, you will be asked to specify the type of delimiter that is being used as shown in Figure 8.5. If the delimiter is not a comma or a tab, click on **Other** and then type the delimiter in the box. The delimiter can be a single character such as a ‘ or a ? or a !. At the bottom of the Delimited ASCII Import window is a box entitled **Change default field names**. If you mark this box, you will be presented with the Field Names window shown in Figure 8.6. If you do not mark this box (so that it is blank), you will skip to the State and County information window shown in Figure 8.7.

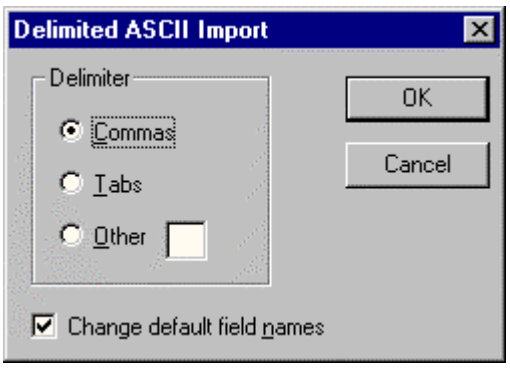


Figure 8.4 Specifying the delimiters for an ASCII delimited file

Generally, an ASCII delimited file does not contain embedded field names. Thus when the ASCII delimited file is read by **BIT** the fields will be called Field001, Field002 and so on. The supplier of the data file should have provided you with documentation that indicates what is contained in each field. The Field Names window in Figure 8.6 allows you to rename the fields in your database so that they are easier to keep track of. In this example, the user has already changed the names of the first three fields. To make a change, double click on the field name so that it is highlighted, then type in the new name. When you have changed the desired fields (you do not have to name all fields), click the **OK** button to save the changes.

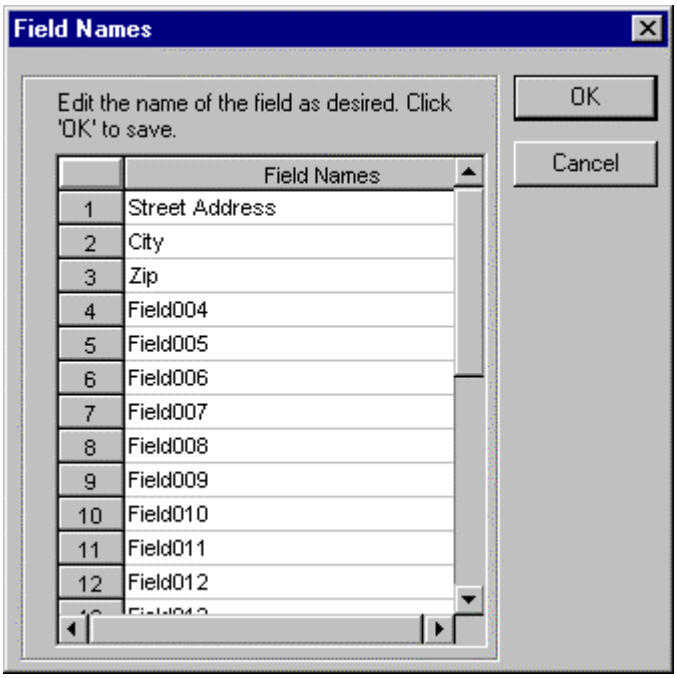


Figure 8.5 Changing the field names in an ASCII delimited file

The next task is to indicate which state and county pertains to your data file (see Figure 8.7). As discussed earlier, the **BIT** can operate on only one county at a time. Therefore your data file must contain data from only one county. The information from this step is used to convert census tract data into a format that can be used by **HAZUS**. The census

tract for the first record in Figure 8.3 is "121.00". The **BIT** will add the state and county codes to this and convert the census tract number to 0607512100, indicating that this census tract is in San Francisco County. When you click the **OK** button you will be ready for the next step of mapping fields (see Section 8.3).

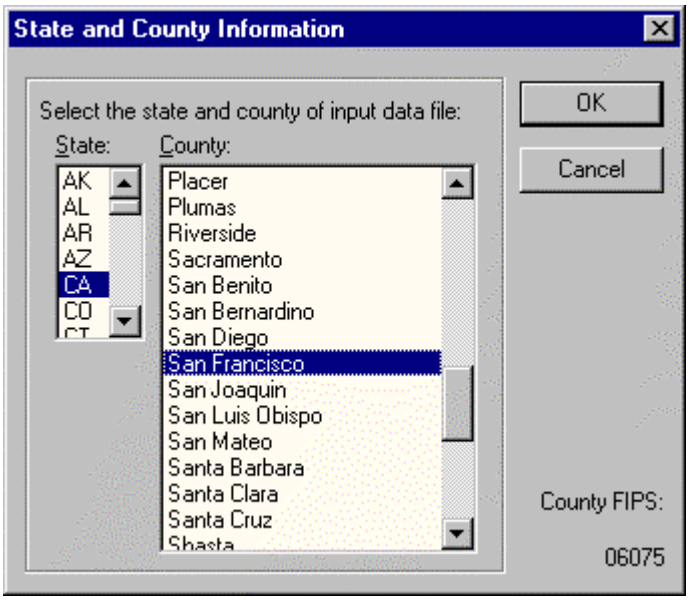


Figure 8.6 Indicating the state and county of the property data file

8.3.2 Importing a *.dbf Database

A file that is in a *.dbf format does not require some of the steps that are required for an ASCII delimited file. Simply specify the file name as shown in Figure 8.7. You will then be presented with the state and county information window shown in Figure 8.6. Complete this information, click **OK**, and you will be ready for mapping fields (see Section 8.3).

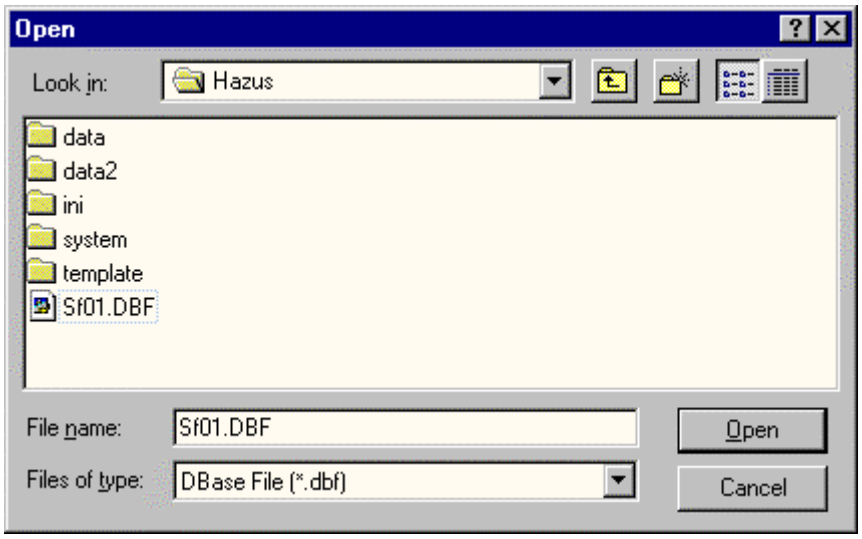


Figure 8.7 Specifying a *.dbf input file in the building-data import tool

8.3.3 Importing a Fixed Length Field Database

A fixed length field file is one that uses a specified number of characters to represent the data in each field. Some fields may be one character long, while other fields may contain 25 characters. Two records from a fixed length field file are shown in Figure 8.8. This is the same data as the comma-delimited records shown in Figure 8.2.

```
521-525 Main St      Anytown    941021102 121.00Store Building        4195
1 2 883263 16 79    880720CConcrete   Stucco    Concrete   Steel      Flat
Built-up       Average    $357    0284-000

332 North St         Anytown    941022607 125.00Apartment            16030
6 24341314 23 7272 830404CConcrete   Concrete Concrete   Concrete   Flat
Tar & Gravel    Fair       $17     0333-001
```

Figure 8.8 Two records from a fixed length field file

Each record in Figure 8.8 spans three lines and the blank spaces indicate missing information or fields that are not completely full. For example, the address field is the first field in the database and the city name is the second. The address field is 20 characters long, but the address in the first record is only 15 characters, thus there are five blank spaces before the city name. Sometimes blank spaces are indicated by something other than a space. For example, all blanks could be filled with the number 8. This is information you need to get from the supplier when you acquire the data. It is important to understand that the **BIT** can recognize a file as a fixed length field file only if you specify the extension of the filename as .txt. To open the file, use the **Open** window shown in Figure 8.9 and specify the file type as **Fixed-length File (*.txt)**.

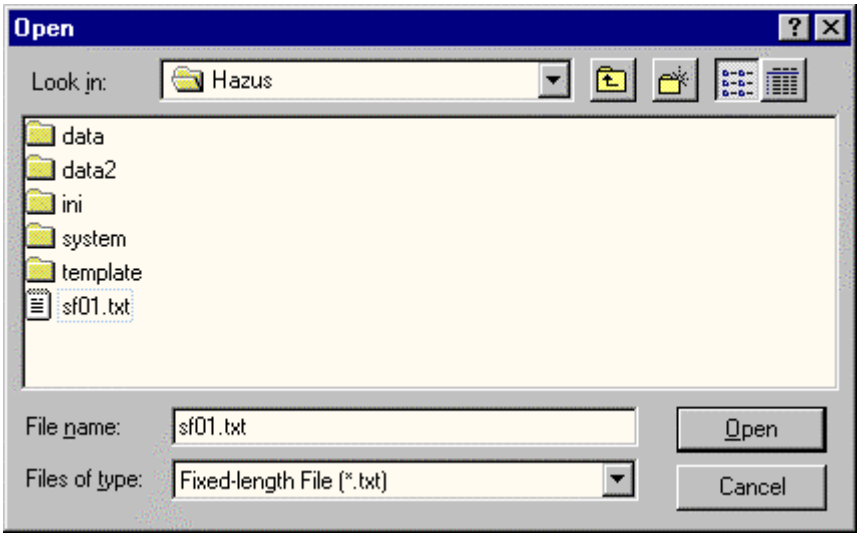


Figure 8.9 Specifying a *.txt input file in the building-data import tool

When **BIT** reads a fixed-length field file, the fields will be labeled Field001, Field002 etc., since the file does not contain embedded field names. The supplier of the data file should have provided you with documentation that indicates what is contained in each field and how long each field is. The Fixed Length File Import Options window in Figure 8.10 is used to name and define the lengths of the fields in your database. In this

example, the user has already defined the names and lengths of the first three fields and is in the process of defining the fourth.

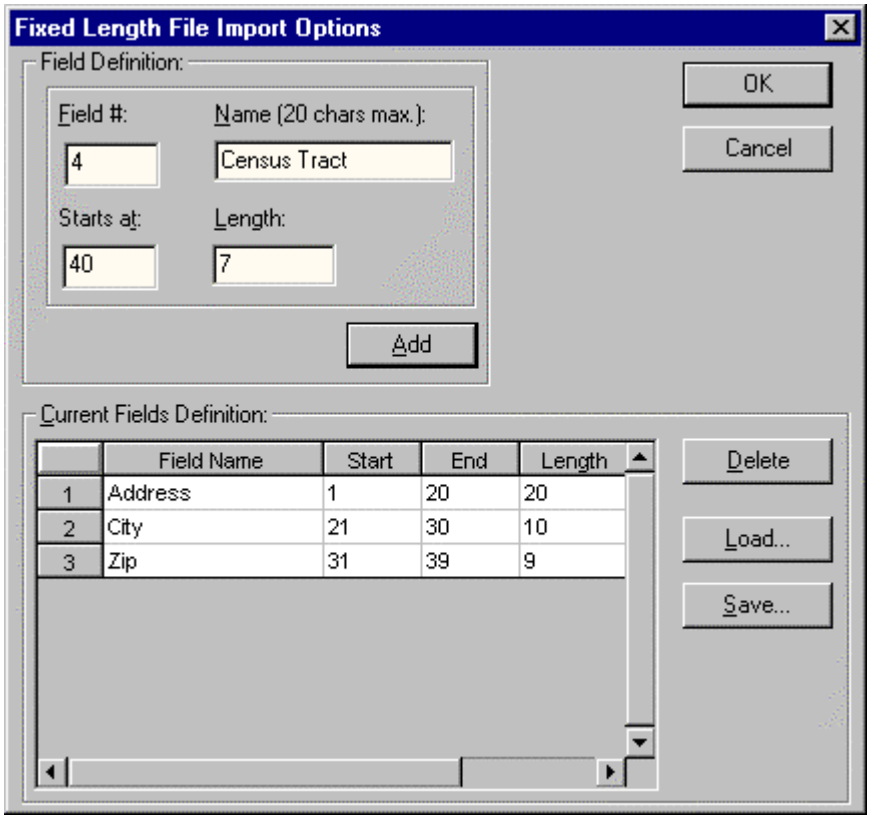


Figure 8.10 Defining and naming fields for a fixed length field file

To define a field, type in the field name and then indicate the length of the field. Click on the **Add** button and the field will be added to the **Current Fields Definition** box. If you realize you have made a mistake after the definition has been added to the **Current Fields Definition** box, highlight the field and click on the **Delete** button. Redefine the field and add it back. When you have defined all of the fields, click the **OK** button to move to the next step. It is important to note that you must define fields for all of the characters in a record in order for the **BIT** to correctly process the database.

Defining all of the fields for long records requires quite a lot of typing. You may want to save the field definitions using the **Save** button. Then if you import any other data files with the same format, you can load the previously established field definitions using the **Load** button.

After you have finished defining the fields and you have clicked **OK**, you will need to indicate the state and county of the database using the window shown in Figure 8.11. The explanation for this step is found in Section 8.2.1.

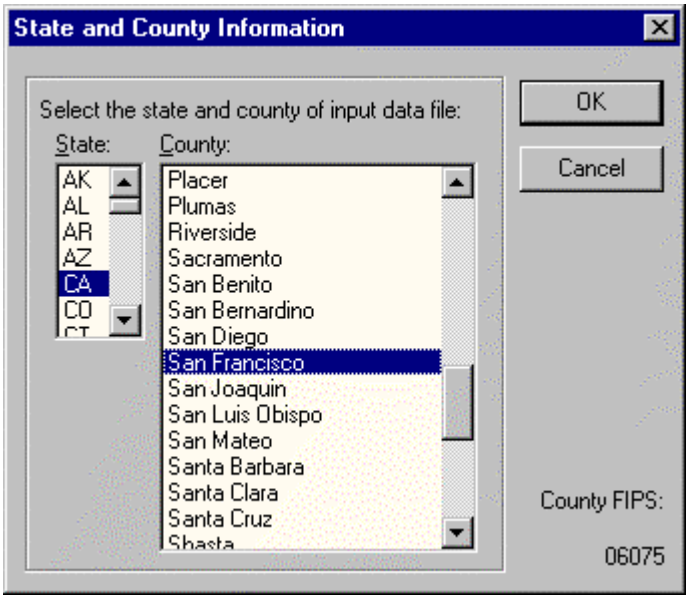


Figure 8.11 Indicating the state and county of the property data file

8.4 Mapping Fields

After having specified the input file, you will need to map the fields in your database (the source) to the fields used in the **HAZUS** database (the target database). The steps for importing data and creating occupancy to model building type relationships must be completed in the numbered sequence. The labels for steps that are not yet available to you will appear in light gray. To start this step, click on the **Map Fields** button in the main **BIT** menu (see Figure 8.12).

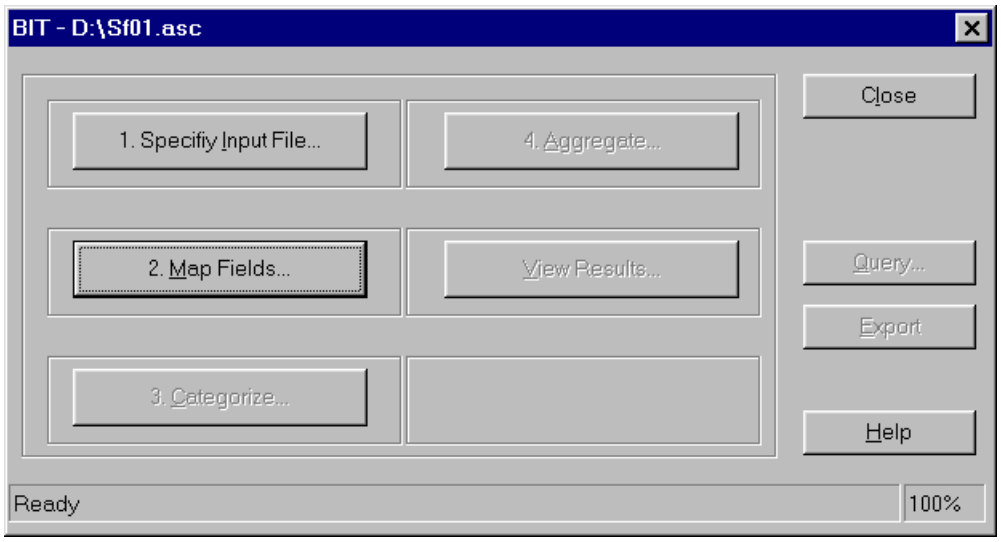


Figure 8.12 Starting the field mapping step from the BIT main menu

Since the **BIT** is used to develop occupancy to model building type relationships for your region, the most important information to capture is the occupancy, structural type, square footage and height of your buildings. However, the database you create can have

as many fields as you want, allowing you to maintain many types of data. Using the mapping tool outlined in this section, you can be certain that all of the databases you maintain will be in a standard format.

The mapping window shown in Figure 8.13 is used to map the fields in your database (the source) to the fields used in **HAZUS** (the target database). The source-database fields do not have to be in the same order nor do they have to have the same names as the target-database fields. For example, in Figure 8.13 the occupancy types are in a field called "OCCUPANCY" in the source database whereas the field that contains this information in the target database is called "OCC_TYPE". When you click on the target field name, a short definition of the field appears right below the list of target fields.

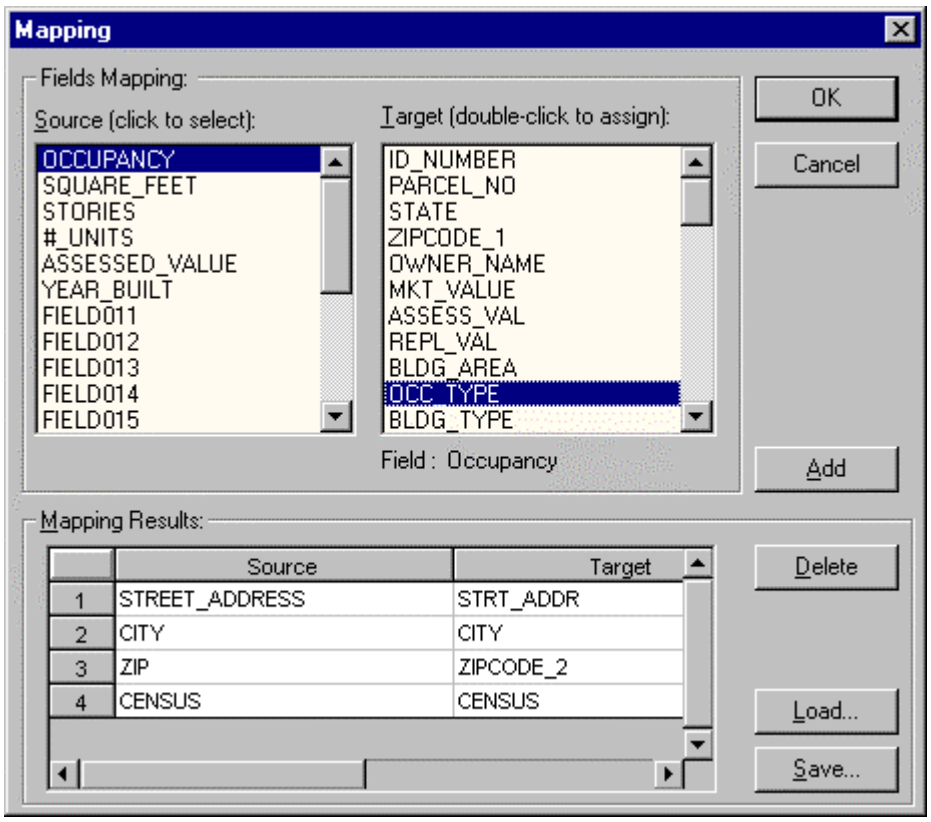


Figure 8.13 Defining a mapping scheme from the source database to the target database in the BIT

To define the desired mapping, simply click on a field name in the source database (e.g. OCCUPANCY) and the corresponding field name in the target database (e.g. OCC_TYPE) and then click on the **Add** button. After each time you perform this operation the mapping you have defined will appear in the **Mapping Results** box at the bottom of the window. At the same time, these fields will disappear from the **Fields Mapping** box at the top of the window. If you make a mistake, click the **Delete** button and the last mapping pair you have defined will be undone. In this example the user has already defined four relationships and is in the process of defining a fifth. When you have completed mapping all of the fields, click on the **OK** button, wait a moment, and your database will be reconfigured into the standardized format. At the end of this step a

file with the same name as your original file and the extension .TG1 will be created. Your original file will remain unchanged. NOTE: You do not have to map all of the fields from the source database; however, any fields you do not map will not be imported into the target database. There are key fields that must be mapped without which you won't be able to proceed with the mapping. The **BIT** tool will prompt you with the key field (s) that you missed mapping once you try to click the OK button to move on to the next step. An example of this window is shown in Figure 8.13. The window also includes the list of the "must mapped" fields for the **BIT** tool.

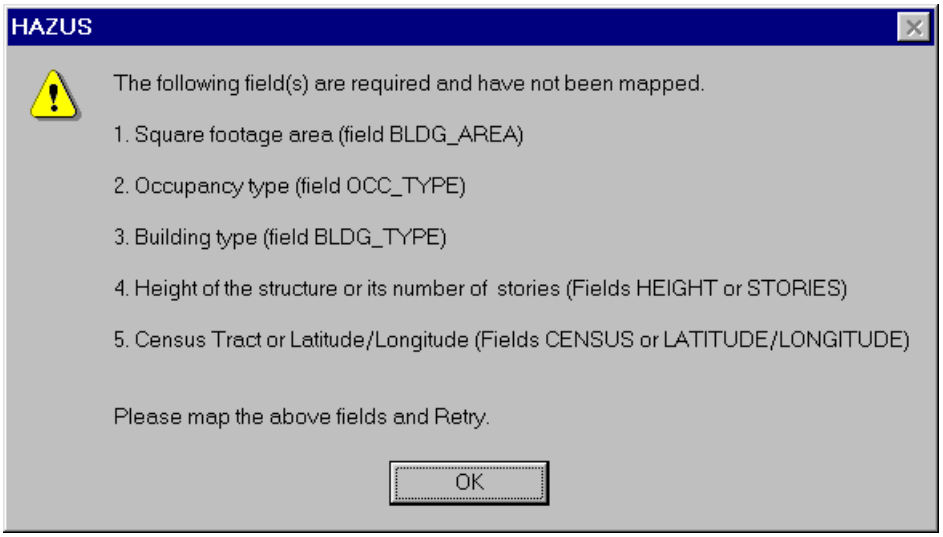


Figure 8.14 An example of a warning message in case you miss mapping key field(s)

It is possible you have several databases with the same format and you would like to save the mapping that you have just defined. Before you click the **OK** button, click the **Save** button in Figure 8.12. A save window will appear and you will need to enter a name for the saved mapping scheme. Retrieve the saved mapping scheme by clicking on the **Load** button in Figure 8.12.

8.5 Categorizing Data

The next step in creating standardized data formats is to convert the data to the classification systems defined in Appendix A. For example, your database may use the term "wood" for low-rise wood frame construction whereas this would be classified as a W1 model building type in **HAZUS**. Thus, records with structural type "wood" in the source database need to be converted to "W1" in the target database. To do this step, click on the **Categorize...** button shown in Figure 8.15. At the end of this step a new file will be created. It will have the same name as your original file and a new extension: .TG2. This database is the same as the *.TG1 database except that all of the replacements you have requested have been made.

Note that at this point the query tool has been enabled. A discussion of the query tool is found in Section 8.8

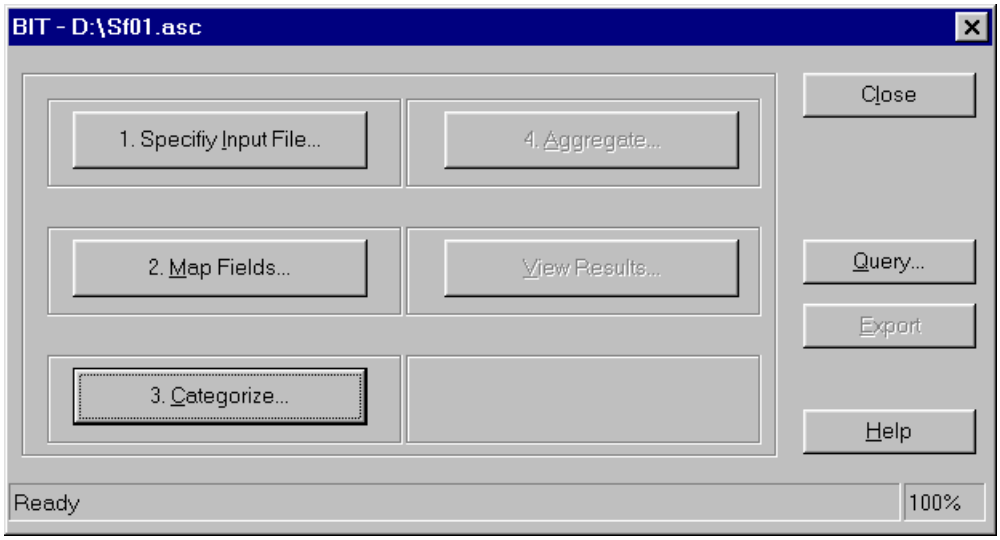


Figure 8.15 Starting the categorize function of the BIT

You have the option to select which fields of data you want to categorize (see Figure 8.16). It is likely that none of your data will be in the standardized format and you will want to select all four options (Stories, Year Built, Occupancy Types and Building Types). To select the items, simply click on them. When you are finished, click the **OK** button.

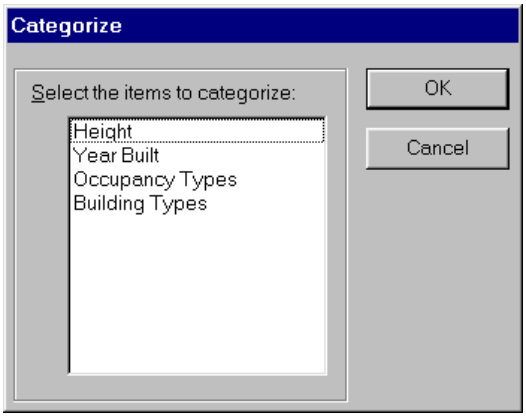


Figure 8.16 Selecting which fields you want to categorize

8.5.1 Categorizing Number of Stories Data

HAZUS lumps groups of buildings into low, medium and high-rise structures. Thus ultimately, any building with one to three stories height will be classified as low rise. If your database uses numbers to specify the height of the building in feet, the **BIT** will automatically convert the height to low, medium or high-rise. Blank fields will be classified as unknown. If the building height that you have is in non-feet units, you can use the conversion factor to convert the data to feet. If on the other hand the database that is being used has characters or words for number of stories, then you will need to define a mapping scheme to convert your data to the standardized format. The window in Figure 8.17 is used to indicate which of these situations apply to your data.

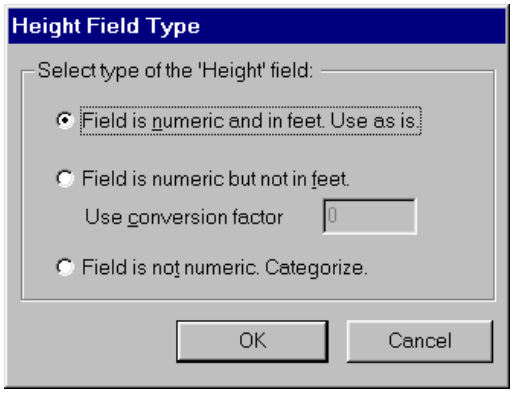


Figure 8.17 Indicating what type of building story data you have

If you click on **Field is non-numeric. Categorize**, then press **OK**, the window in Figure 8.18 is displayed allowing you to define a mapping from your database to the standardized format. As indicated by the text labeled **# occurrences** and **Out of,** there are ten records in this example, and all of the records have a blank in the field containing the number of stories data. The user has mapped the blank to "Unknown". As with other mapping windows, after you have defined each mapping, click on the **Add** button and the mapping will appear in the **Results** portion of the window. If you make a mistake, use the **Delete** button.

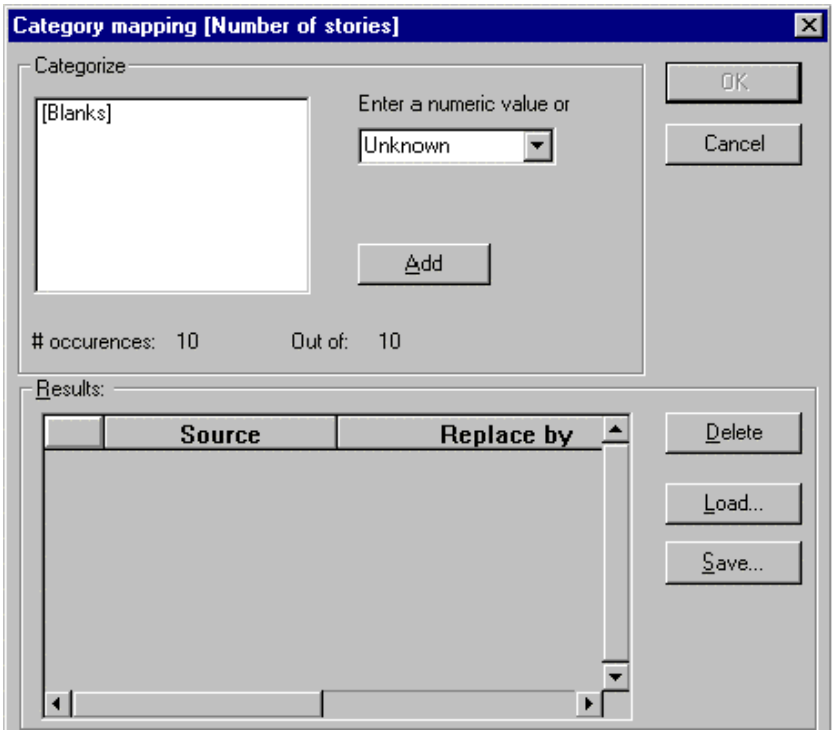


Figure 8.18 Categorizing number of stories data

To save your data mapping scheme, click on the **Save...** button. Use the window shown in Figure 8.19 to name the mapping scheme. A scheme for mapping number of stories

will have an .ssl extension, whereas a scheme for mapping building height will have an .hsl extension.

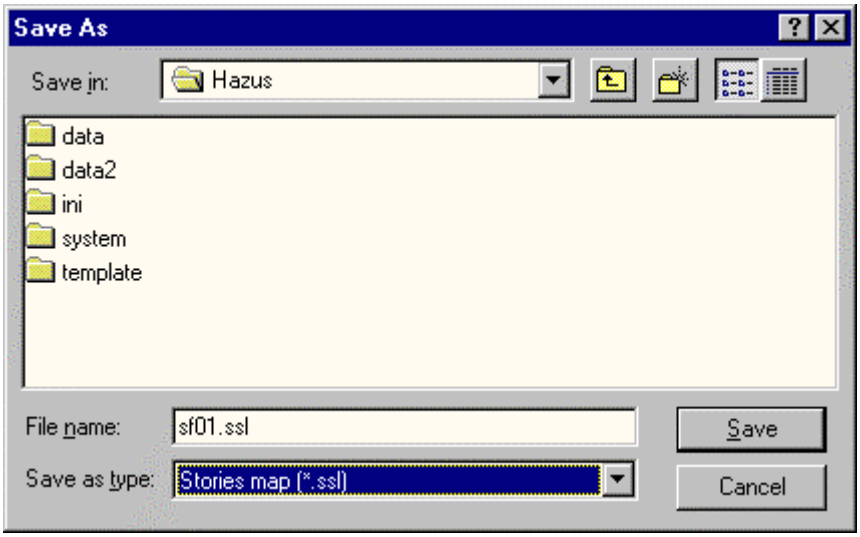


Figure 8.19 Saving number of stories categories

8.5.2 Categorizing Year Built Data

HAZUS lumps buildings into three age groups: pre-1950, 1950-1970 and post-1970. Occupancy to model building type relationships is developed for each of these three groupings. Year-built data is found in a variety of formats in assessor's files and other commercially available property files. It is most common to find the year built expressed in a two-digit format, such as 95, or in a four-digit format, such as 1995. However, it is possible that other formats could be used such as old, moderate and new. The **BIT** has the flexibility to read any of these formats by selecting the appropriate buttons in Figure 8.20. Perhaps most problematic is how to deal with a zero. A zero can mean that a structure was built in 1900 or it can mean that the data is unknown. You may have to ask the supplier of the data how to interpret the occurrence of a zero in the data.

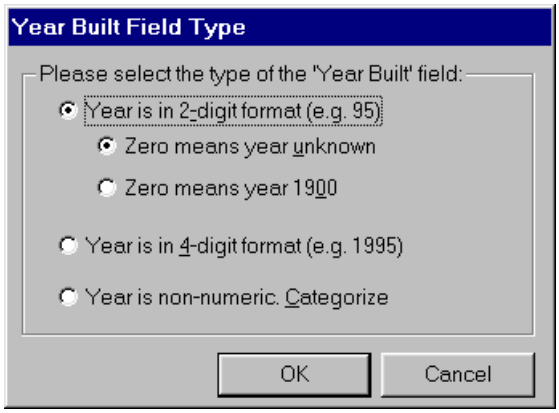


Figure 8.20 Categorizing year built data

8.5.3 Categorizing Occupancy Class Data

In this step you will be required to map the occupancies found in the source database to the standardized occupancies defined in **HAZUS** (See Appendix A, Table A.3). All of the 28 specific occupancy classes found in Table A.3 are listed in the **Target** list box found in Figure 8.21. In addition to the specific occupancy classes, you will find five general occupancy classes (Residential, Commercial, Industrial, Government, Education) and the class "Unknown". General occupancy classes are in all upper-case letters. Some property databases contain very limited information about occupancy; for example, labels such as residential, commercial, and industrial. In this case you will need to use the general occupancy classes for categorizing occupancy.

To define a mapping, click on an occupancy in the **Source** list box and then double click on the corresponding standardized occupancy in the **Target** list box. You can map multiple occupancies at the same time by highlighting the occupancies in the Source list box that correspond to a single standardized occupancy. For example, in Figure 8.21 the user has already highlighted "apartment", "condominium", "hotel", and "duplex" in the **Source** list box and then double clicked on "Multi Family Dwelling" in the **Target** list box. This resulted in the four separate mappings found in the **Mapping Results** box. If you find you have made a mistake any time during this process, simply click on the incorrect mapping in the **Mapping Results** box and click on the **Delete** button. Redefine the correct mapping for that occupancy and continue. When you have completed the mapping for all categories in the source database, click the **OK** button.

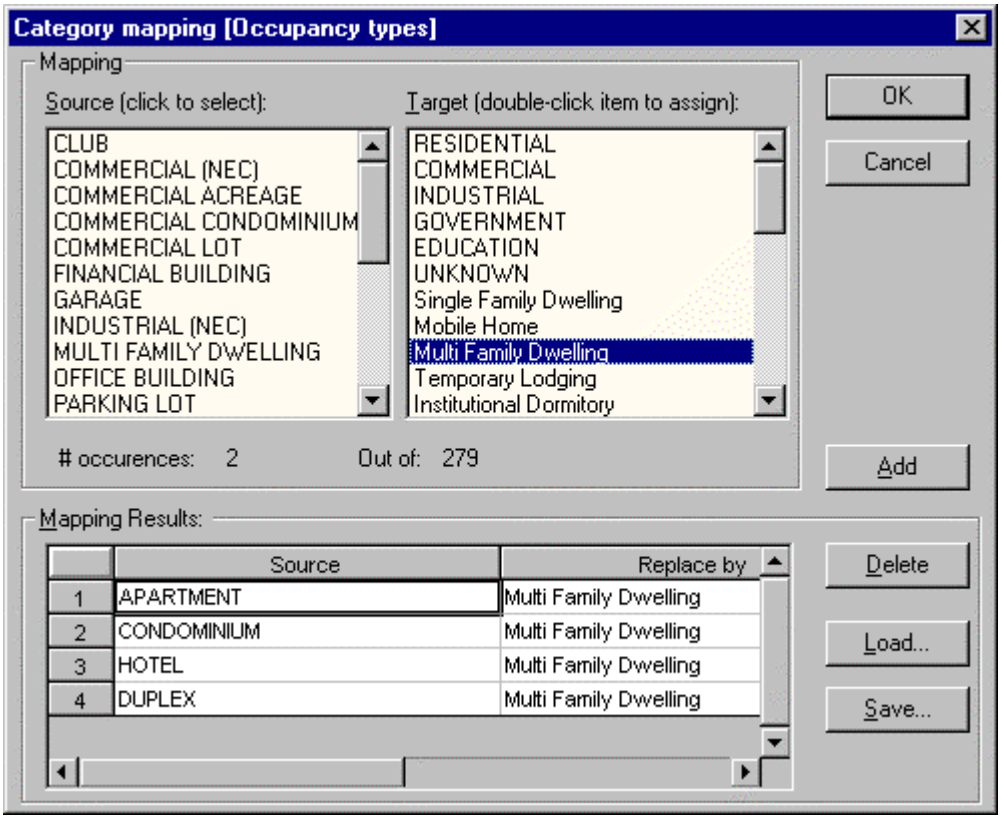


Figure 8.21 Categorizing occupancy class data

Categorizing occupancy class data can be somewhat tricky and can require judgment on your part. Some of the occupancy classes in the property file may not fit perfectly into **HAZUS** classifications. For example, you may find a class such as "Office & Residential" in your database that fits into both the RES3 "multi-family dwelling" class and the COM4 "Financial/Professional/Technical Services" class. You will have to use your judgment in deciding which standardized class best typifies this mixed occupancy. Another problem you may find is that source-database occupancy classes do not always provide a correct description of the property. For example, parking lot, residential lot or vacant lot would imply that these properties have no structures on them. However, in many cases in the sample database used here, there were buildings on these types of properties. You should not be surprised to find that certain occupancies such as universities, institutional housing and government services, to name a few may, be completely absent from your database. As noted in Section 5.1.2, property databases rarely provide detailed information on tax-exempt properties.

As with other mappings defined in the **BIT**, you have the option to save the occupancy class mapping for use on other files. To save the mapping, click on the **Save...** button before clicking **OK**. The occupancy mapping file will be saved with an .osl extension as shown in Figure 8.22. To use the mapping in the future, click on the **Load...** button in Figure 8.21.

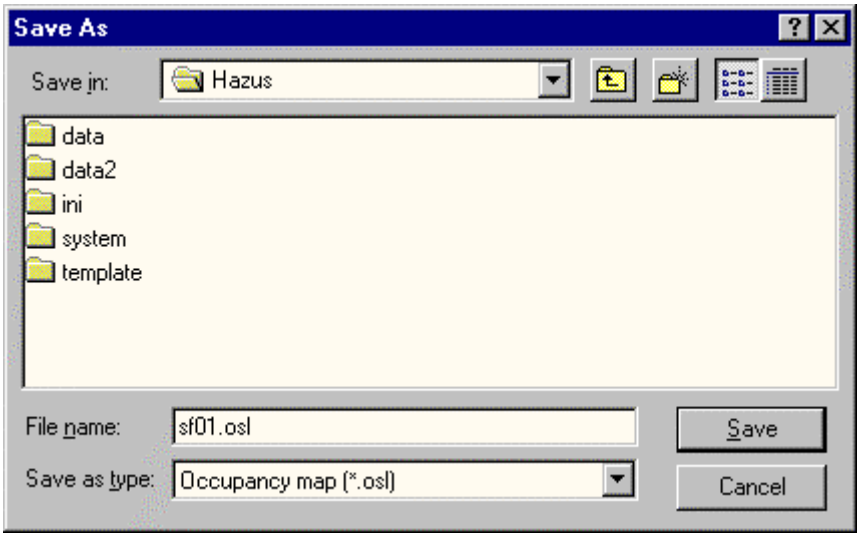


Figure 8.22 Saving an occupancy mapping scheme

8.5.4 Categorizing Building Type Data

In this step you will be required to map the structural types found in the source database to the model building types defined in **HAZUS** (See Appendix A, Table A.2). All of the 16 general building types found in Table A.2 are listed in the **Target** list box found in Figure 8.23. In addition to the general model building types, you will find four basic building material types (Wood, Steel, Concrete, Masonry) and the class "Unknown". Basic building material types are in all upper-case letters. Many property databases contain very limited information about the structural system used, and the categories used are often based on fire safety information. For example, in this sample database shown in

Figure 8.2 and 8.8, category C contains brick, tilt-up and formed concrete construction. The user has chosen to map category C to masonry. Clearly, this will introduce uncertainty into the occupancy to model building type relationships that are produced by the **BIT**. It is rare that you will find a property database that will give you enough information to define reliable mappings to all general building types.

To define a mapping, click on a building type in the **Source** list box and then double-click on the corresponding standardized building type in the **Target** list box. You can map multiple building types at the same time by highlighting (clicking on) all of the building types in the **Source** list box that correspond to a single standardized building type. For example, in Figure 8.23 the user highlighted "A" and "S" in the **Source** list box and then double clicked on "STEEL" in the **Target** list box. This resulted in the two separate mappings found in the **Mapping Results** box. If you find you have made a mistake any time during this process, simply click on the incorrect mapping in the **Mapping Results** box and click on the **Delete** button. Redefine the correct mapping for that building type and continue.

When you have completed the mapping for all categories in the source database, click the **OK** button. At this point the **BIT** will substitute the standardized categories for the original categories in the source database. Depending on the size of the database this will take a few minutes to more than an hour.

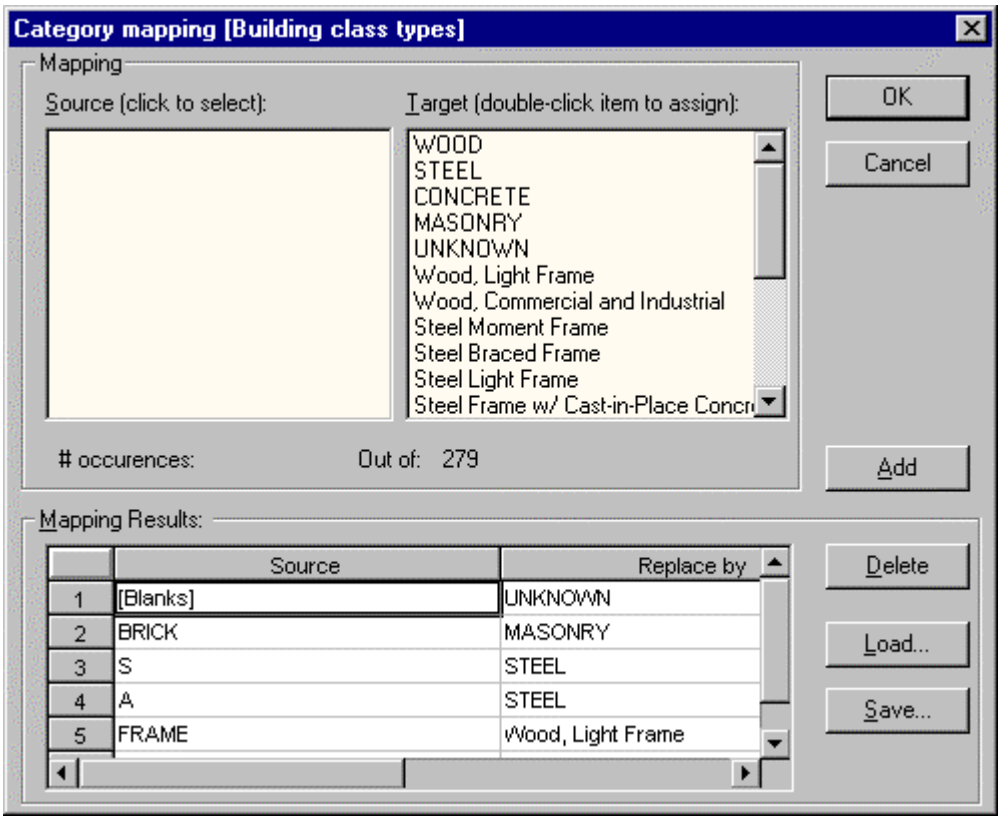


Figure 8.23 Categorizing building type data

As with other mappings defined in the **BIT**, you have the option to save the building type mapping for use on other files. To save the mapping, click on the **Save...** button before clicking **OK**. The building type mapping file will be saved with a .bsl extension as shown in Figure 8.24. To use the mapping in the future, click on the **Load...** button in Figure 8.23.

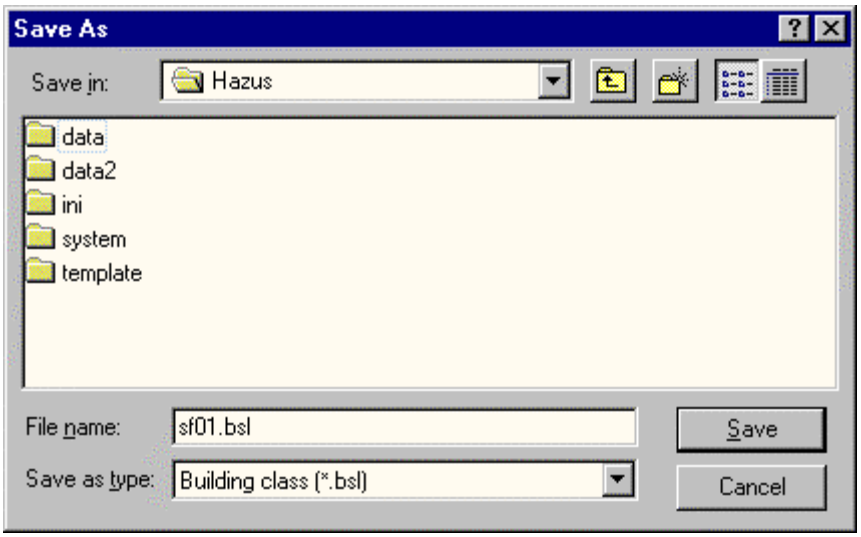


Figure 8.24 Saving a building type mapping scheme

8.6 Aggregating the Database Statistics

At this point the **BIT** is ready to create the occupancy to model building type relationships for each census tract. Click on the **Aggregate** button (shown in Figure 8.25) and wait. When the aggregation is done you will be able to view the results using the **View Results** button.

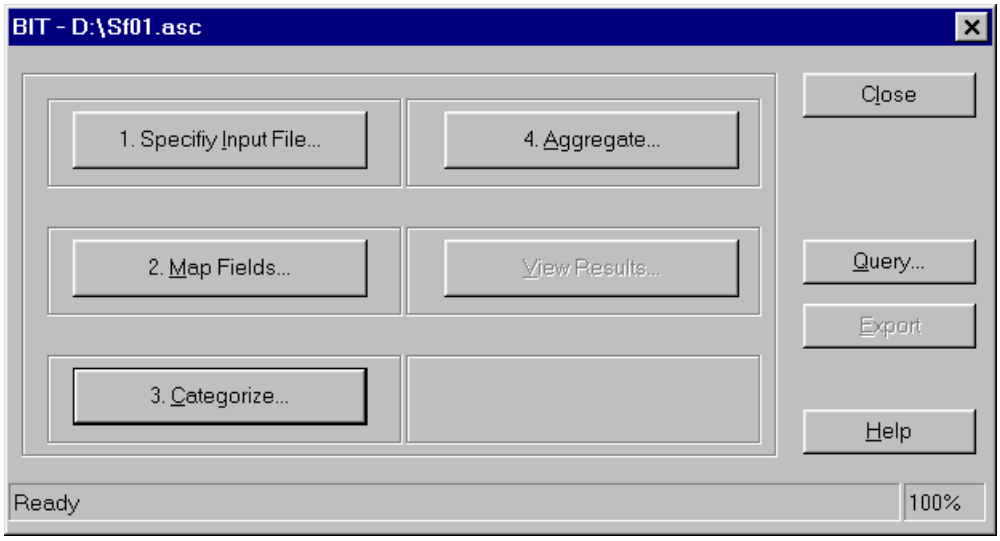


Figure 8.25 Starting the aggregation utility

If for some reason you have changed your database in some way and need to run the aggregate utility again, you will be asked if you want to overwrite the files you created previously. An example of this window is found in Figure 8.26.

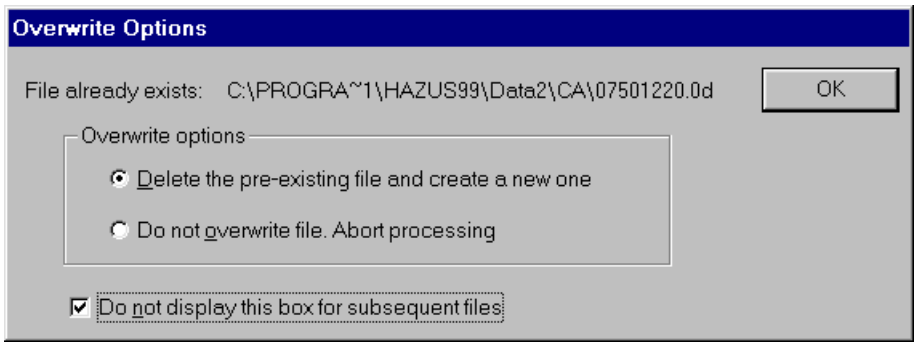


Figure 8.26 Overwriting previously developed occupancy to model building type relationships

8.7 Viewing the Results

To view the updated databases and the occupancy-to model-building-type mapping schemes, click on the **View Results...** button shown in Figure 8.27.

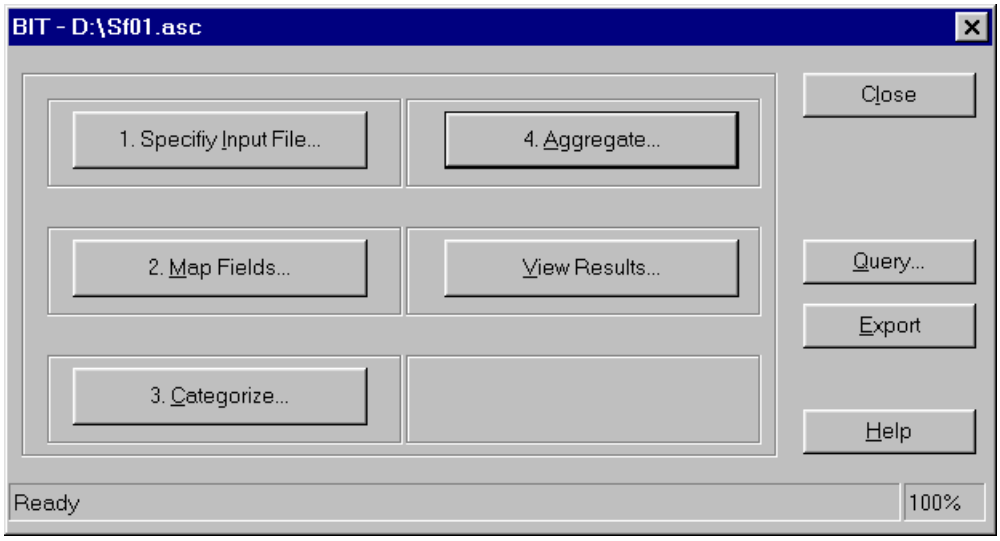
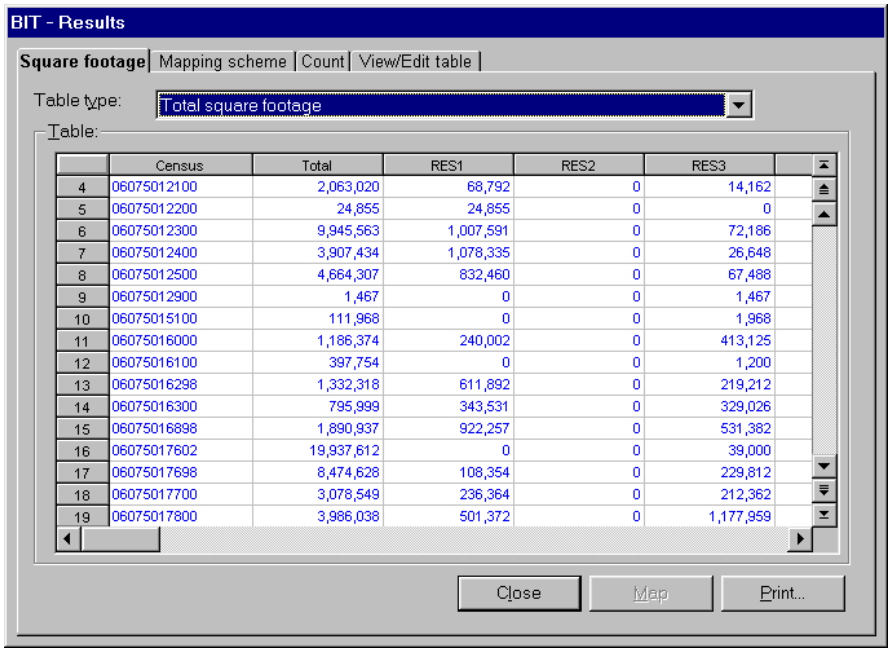


Figure 8.27 Starting the View Results utility

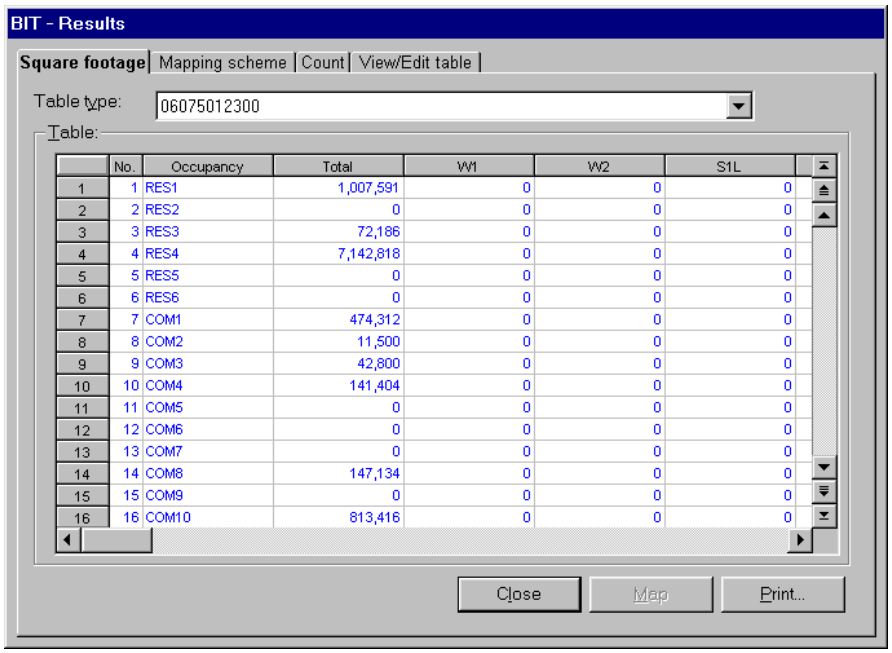
8.7.1 Viewing Square Footage

The **BIT** gives you the option to view the total square footage for each occupancy type and census tract as shown in Figure 8.28. Alternatively, for any individual census tract you can view the square footage for each occupancy and building type as shown in Figure 8.29. To access a particular census tract, use the **Table type:** list box near the top of the **BIT - Results** window in Figure 8.29 and click on the census tract of interest to you.



	Census	Total	RES1	RES2	RES3
4	06075012100	2,063,020	68,792	0	14,162
5	06075012200	24,855	24,855	0	0
6	06075012300	9,945,563	1,007,591	0	72,186
7	06075012400	3,907,434	1,078,335	0	26,648
8	06075012500	4,664,307	832,460	0	67,488
9	06075012900	1,467	0	0	1,467
10	06075015100	111,968	0	0	1,968
11	06075016000	1,186,374	240,002	0	413,125
12	06075016100	397,754	0	0	1,200
13	06075016298	1,332,318	611,892	0	219,212
14	06075016300	795,999	343,531	0	329,026
15	06075016898	1,890,937	922,257	0	531,382
16	06075017602	19,937,612	0	0	39,000
17	06075017698	8,474,628	108,354	0	229,812
18	06075017700	3,078,549	236,364	0	212,362
19	06075017800	3,986,038	501,372	0	1,177,959

Figure 8.28 Viewing total square footage for each occupancy type



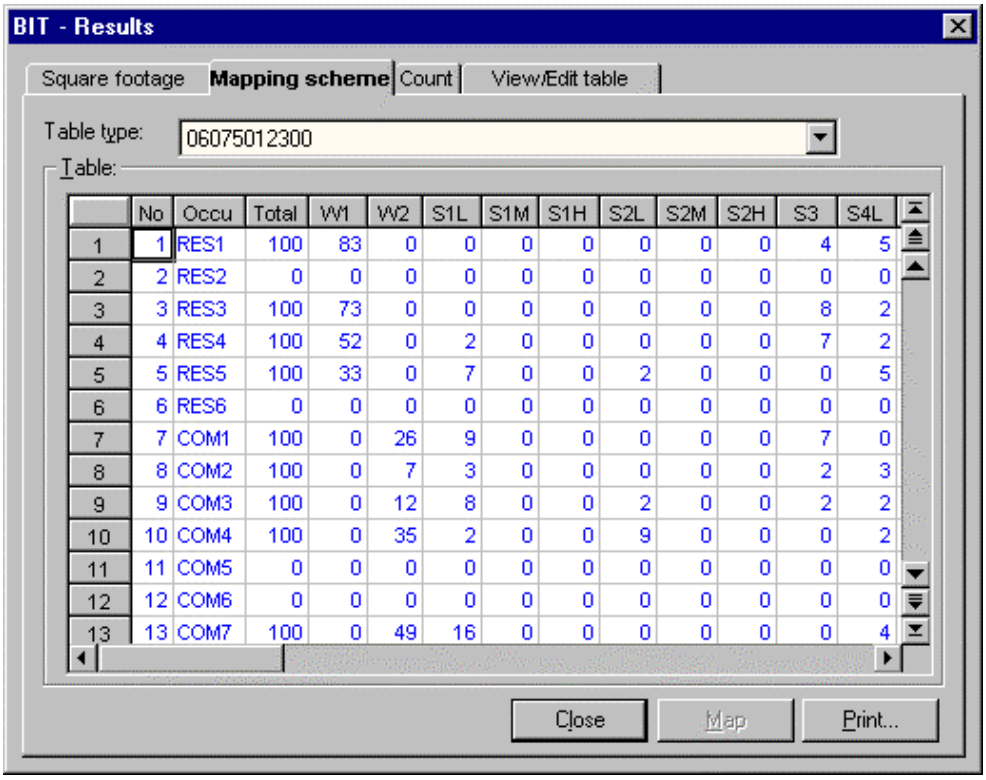
	No.	Occupancy	Total	W1	W2	S1L
1	1	RES1	1,007,591	0	0	0
2	2	RES2	0	0	0	0
3	3	RES3	72,186	0	0	0
4	4	RES4	7,142,818	0	0	0
5	5	RES5	0	0	0	0
6	6	RES6	0	0	0	0
7	7	COM1	474,312	0	0	0
8	8	COM2	11,500	0	0	0
9	9	COM3	42,800	0	0	0
10	10	COM4	141,404	0	0	0
11	11	COM5	0	0	0	0
12	12	COM6	0	0	0	0
13	13	COM7	0	0	0	0
14	14	COM8	147,134	0	0	0
15	15	COM9	0	0	0	0
16	16	COM10	813,416	0	0	0

Figure 8.29 Viewing occupancy and model building type square footage for a particular census tract

8.7.2 Viewing and Using Mapping Schemes

Mapping schemes can be accessed by clicking on the **Mapping Scheme** tab at the top of the **BIT - Results** window. A mapping scheme has been created for each census tract. To access a mapping scheme for a particular census tract, use the **Table type:** list box near the top of the **BIT - Results** window and click on the census tract of interest to you.

The mapping scheme is presented in terms of percentages. For example, for census tract 06075012300 in Figure 8.30, 52% of RES4 are building type W1, 2% are S1L, 7% are S3, 2% are S4L and so on. It should be noted that in developing this mapping scheme, the **BIT** made use of default mappings in cases where no data was available from the property file.



	No	Occu	Total	W1	W2	S1L	S1M	S1H	S2L	S2M	S2H	S3	S4L
1	1	RES1	100	83	0	0	0	0	0	0	0	4	5
2	2	RES2	0	0	0	0	0	0	0	0	0	0	0
3	3	RES3	100	73	0	0	0	0	0	0	0	8	2
4	4	RES4	100	52	0	2	0	0	0	0	0	7	2
5	5	RES5	100	33	0	7	0	0	2	0	0	0	5
6	6	RES6	0	0	0	0	0	0	0	0	0	0	0
7	7	COM1	100	0	26	9	0	0	0	0	0	7	0
8	8	COM2	100	0	7	3	0	0	0	0	0	2	3
9	9	COM3	100	0	12	8	0	0	2	0	0	2	2
10	10	COM4	100	0	35	2	0	0	9	0	0	0	2
11	11	COM5	0	0	0	0	0	0	0	0	0	0	0
12	12	COM6	0	0	0	0	0	0	0	0	0	0	0
13	13	COM7	100	0	49	16	0	0	0	0	0	0	4

Figure 8.30 Viewing a mapping scheme for census tract 06075012300 created by the BIT

To use this mapping scheme, exit the **BIT** and start the **HAZUS** program. Open the study region that includes this census tract. From the **Inventory|General Building Stock| Occupancy Mapping...** menu click on the **Open...** button. You will be presented with the Mapping Scheme files window shown in Figure 8.33. In this example, the user has already imported mapping schemes developed by the **BIT**: 1AV107Y4, 1AV109C4, 1AV109HO…etc. The names of these mappings, constrained by a limitation of eight characters, are difficult to interpret; however, the census tract designation is displayed at the bottom of Figure 8.31 to clarify.

Click on the **Import...** button. Select one or more mapping schemes to import from the window in Figure 8.32. If you want to import a group of census tracts, click on the first census tract number in the group and then hold down the Shift key and click on the last census tract number in the group. To import several census tracts that are not listed consecutively, hold down the Ctrl key and click on the census tracts you want. When you have selected the census tracts, click **OK**. Once you have imported the mapping schemes, follow the instructions in Section 7.3 of this manual.

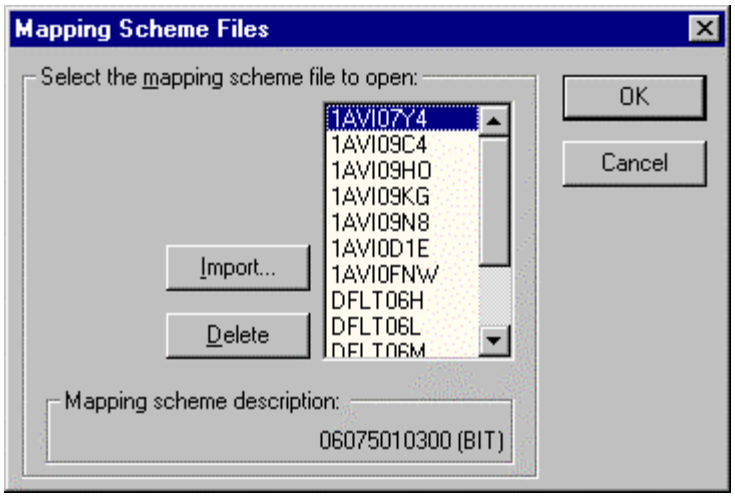


Figure 8.31 Selecting the mapping scheme developed for census tract 06075010300

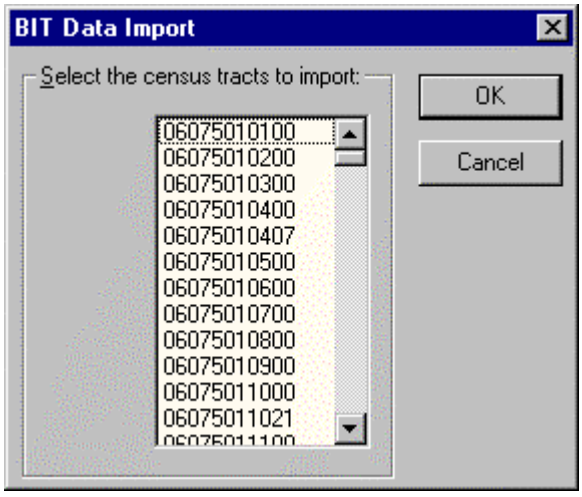


Figure 8.32 Importing mapping schemes developed by the BIT

8.7.3 Viewing Building Counts

By clicking on the **Count** tab at the top of Figure 8.33, for each census tract you can view the number of buildings that were found in the property file for each specific and general occupancy class. This may be helpful in determining the reliability of the mapping scheme. For example, in census tract 06075016300, only one building was identified, thus the resulting mapping scheme will be essentially the default mapping scheme defined in **HAZUS**.

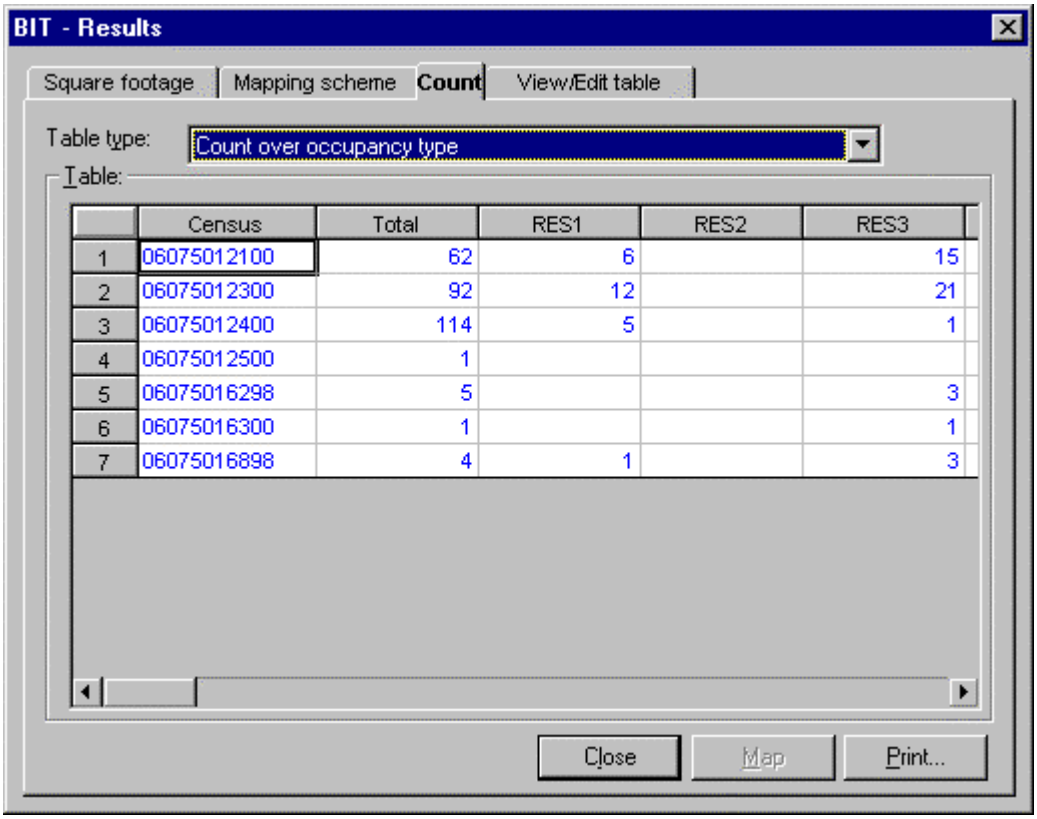
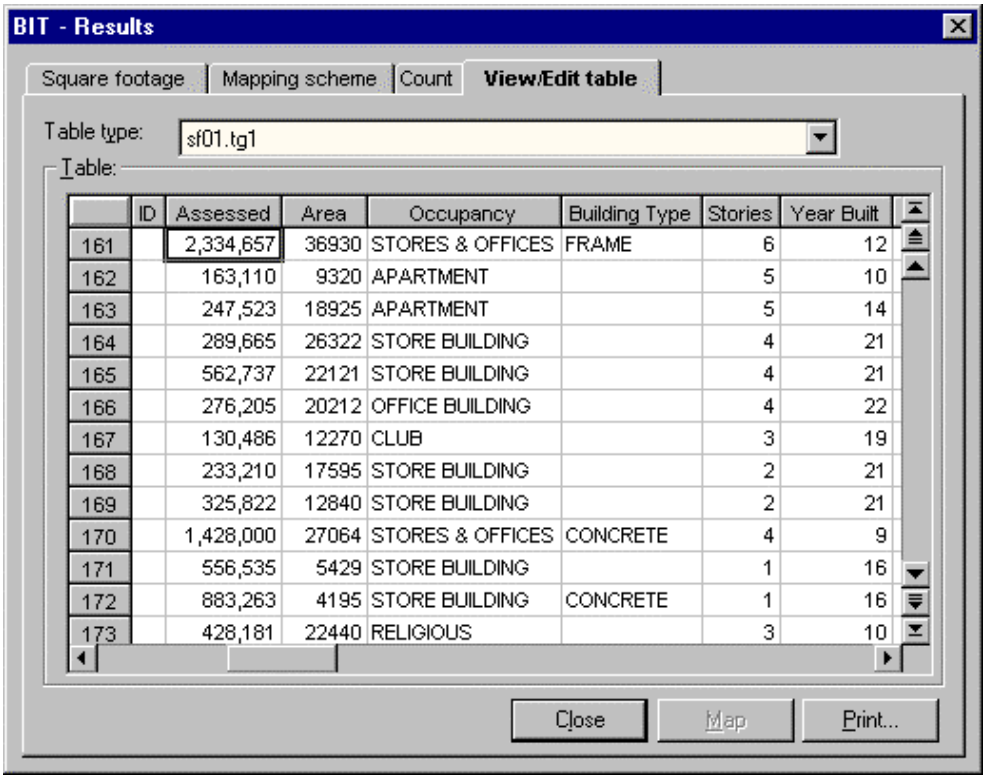


Figure 8.33 Viewing the number of buildings for each occupancy

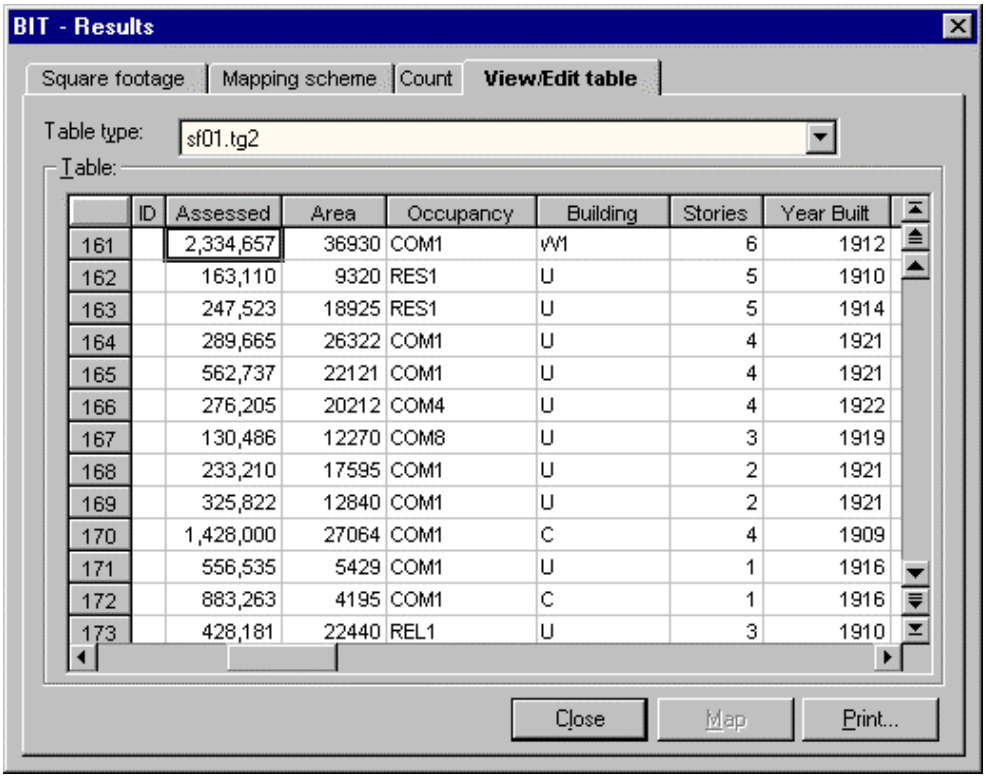
8.7.4 Viewing and Editing Property Files

Clicking on the View/Edit table tab at the top of Figure 8.34 allows you to view the property database that you have imported with the **BIT**. Two files have been created during the process discussed in this chapter: *.tg1 and *.tg2. The *.tg1 file (Figure 8.34) maintains the data exactly as it was in the original property file. The *.tg2 file (Figure 8.35) contains the standardized occupancies, years, types and heights resulting from the substitutions made. Accessing either of these files is achieved by using the **Table type:** list box at the top of Figure 8.34.



	ID	Assessed	Area	Occupancy	Building Type	Stories	Year Built
161		2,334,657	36930	STORES & OFFICES	FRAME	6	12
162		163,110	9320	APARTMENT		5	10
163		247,523	18925	APARTMENT		5	14
164		289,665	26322	STORE BUILDING		4	21
165		562,737	22121	STORE BUILDING		4	21
166		276,205	20212	OFFICE BUILDING		4	22
167		130,486	12270	CLUB		3	19
168		233,210	17595	STORE BUILDING		2	21
169		325,822	12840	STORE BUILDING		2	21
170		1,428,000	27064	STORES & OFFICES	CONCRETE	4	9
171		556,535	5429	STORE BUILDING		1	16
172		883,263	4195	STORE BUILDING	CONCRETE	1	16
173		428,181	22440	RELIGIOUS		3	10

Figure 8.34 Viewing the property database before substitutions have been made



	ID	Assessed	Area	Occupancy	Building	Stories	Year Built
161		2,334,657	36930	COM1	W1	6	1912
162		163,110	9320	RES1	U	5	1910
163		247,523	18925	RES1	U	5	1914
164		289,665	26322	COM1	U	4	1921
165		562,737	22121	COM1	U	4	1921
166		276,205	20212	COM4	U	4	1922
167		130,486	12270	COM8	U	3	1919
168		233,210	17595	COM1	U	2	1921
169		325,822	12840	COM1	U	2	1921
170		1,428,000	27064	COM1	C	4	1909
171		556,535	5429	COM1	U	1	1916
172		883,263	4195	COM1	C	1	1916
173		428,181	22440	REL1	U	3	1910

Figure 8.35 Viewing the property database after substitutions have been made

8.8 Querying Your Database

As noted in Section 8.4, after completing the Map Fields step the query tool is enabled. This tool allows you to search your database to look for particular building types or view summary statistics. The query tool can be used at the Map Fields step to look at the *.tg1 file. In this case you will be looking at the property file before any substitutions of standardized occupancies etc. have been made. Alternatively, it can be run after the Categorize step (on the *.tg2 file) to perform queries using the standardized occupancies and building types.

After you click on the **Query** button (see Figure 8.27), the **Query Statement** window shown in Figure 8.36 will appear. In the window you can choose whether to query the *.tg1 file or the *.tg2 file using the **Table to query:** list box. You can set up any sort of query statement using up to three conditions. The query statements can have equalities and inequalities. In this particular example, the user has a database that uses the letter C to define a building that is either cast-in-place concrete or masonry. She has decided to look for all properties that might be masonry by finding properties where the BLDG_TYPE column contains the letter C.

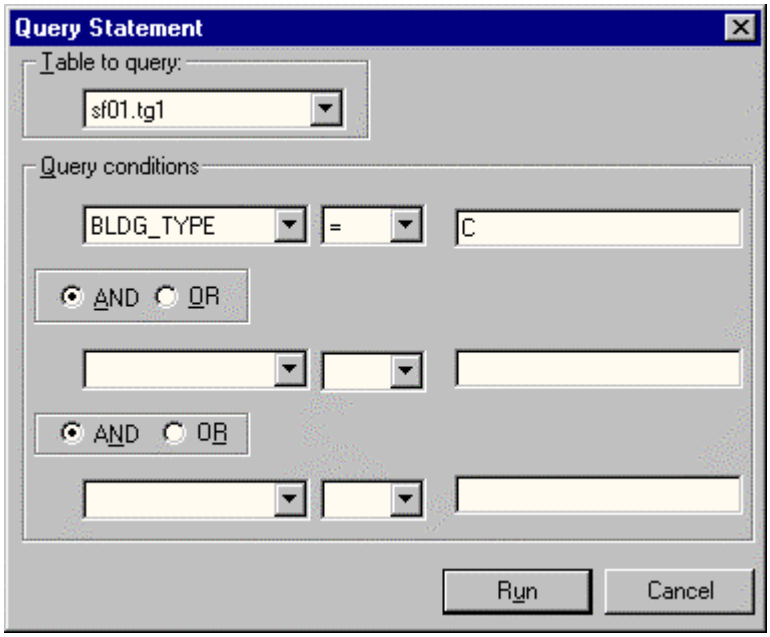


Figure 8.36 Setting up a query statement

The results of the query are found in Figure 8.37. As can be seen at the top of the window, 15 properties were found. The first few columns of the database are displayed and you can scroll to the left or the right to view other columns. By scrolling to the left you would display the property addresses. The important thing to note in this figure is that the occupancy, the number of stories and the year built are in the format of the original property file.

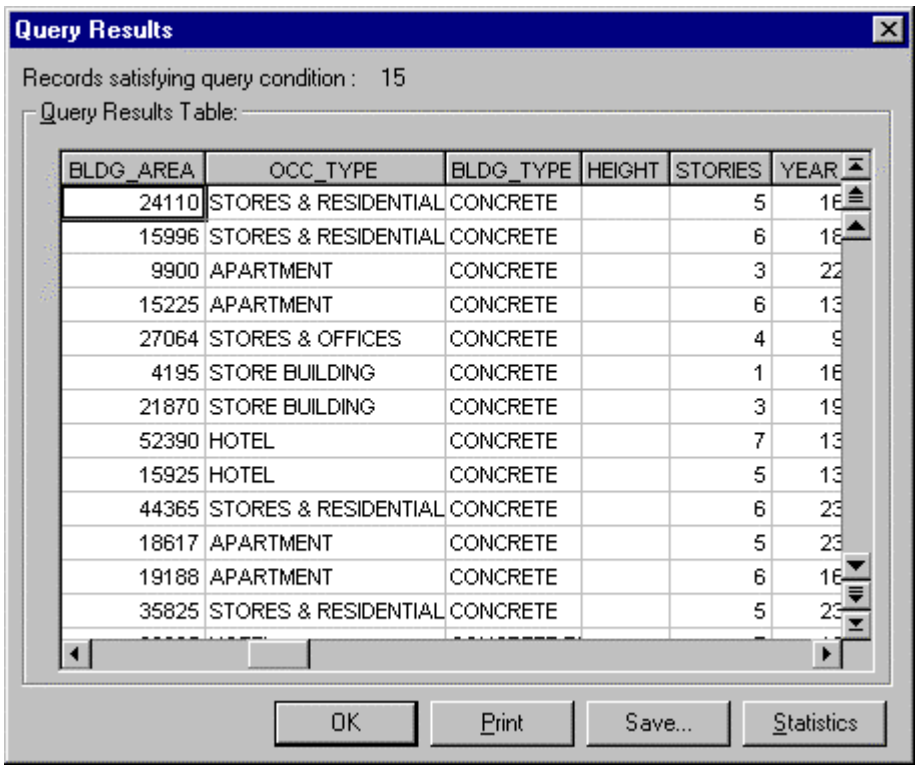


Figure 8.37 Results of query to identify all buildings of type "C"

You can perform statistical analyses on the results of this query by clicking on the **Statistics** button at the bottom of 8.37. You can find out how many square feet of building type "C" are in the property file by highlighting the BLDG_AREA column (clicking on the label) and then clicking on the **Statistics** button. The statistics results are found in Figure 8.38.

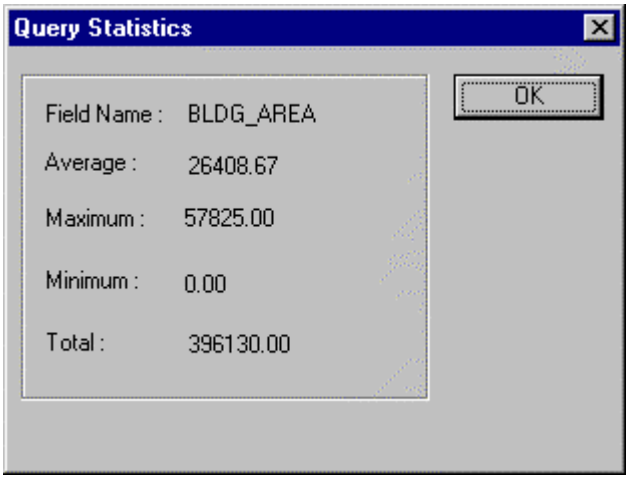


Figure 8.38 Statistics on BLDG_AREA column of query results

The results of a query in a *.dbf file can be saved for use at a later time. To do this click on the **SAVE...** button in Figure 8.37 and name the file in the window shown in Figure

8.39. You can now open this file at any time using Excel, or any database manager that reads a dBase file. It could also be imported into **HAZUS**.

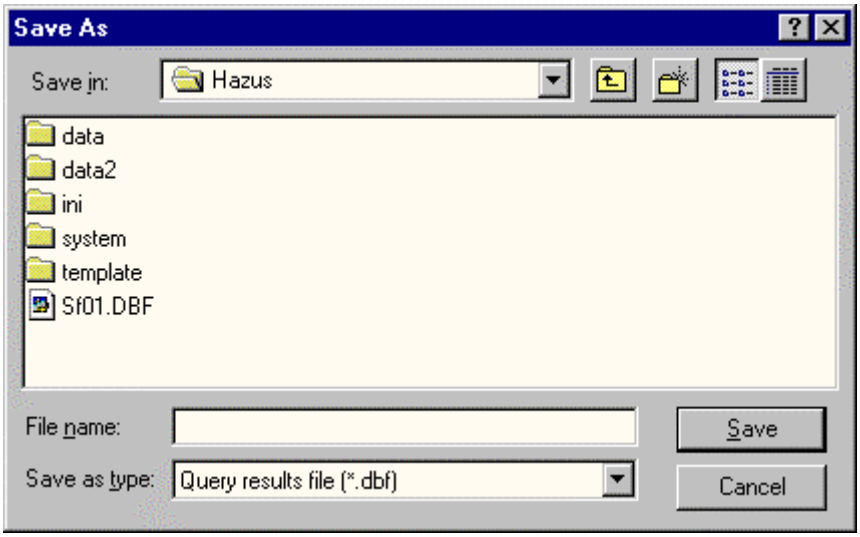


Figure 8.39 Saving a query.

You can also constrain the query to ensure that you are identifying masonry buildings. In addition to buildings with a BLDG_TYPE = C condition, you can also constrain the BLDG_TYPE column to have the word BRICK or the word MASONRY in it. This query statement is shown in Figure 8.40. Since a previous query has already been performed, you will be asked if you want to overwrite the previous query (Figure 8.41). Since the previous query was saved, you will answer yes.

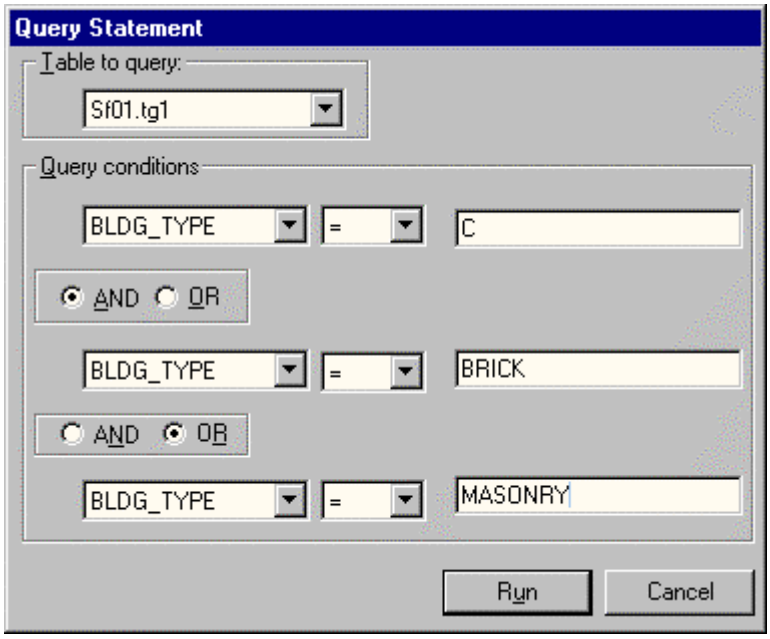


Figure 8.40 Query to identify masonry buildings

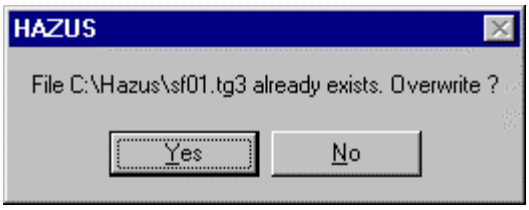


Figure 8.41 Confirmation to overwrite a previous query

The results of the query are found in Figure 8.42. This time only 5 properties were identified. You can see that the building area and occupancy type are two of the types of information that are available. The user can then do sidewalk surveys of these properties, import this database and map these properties or a number of other useful things.

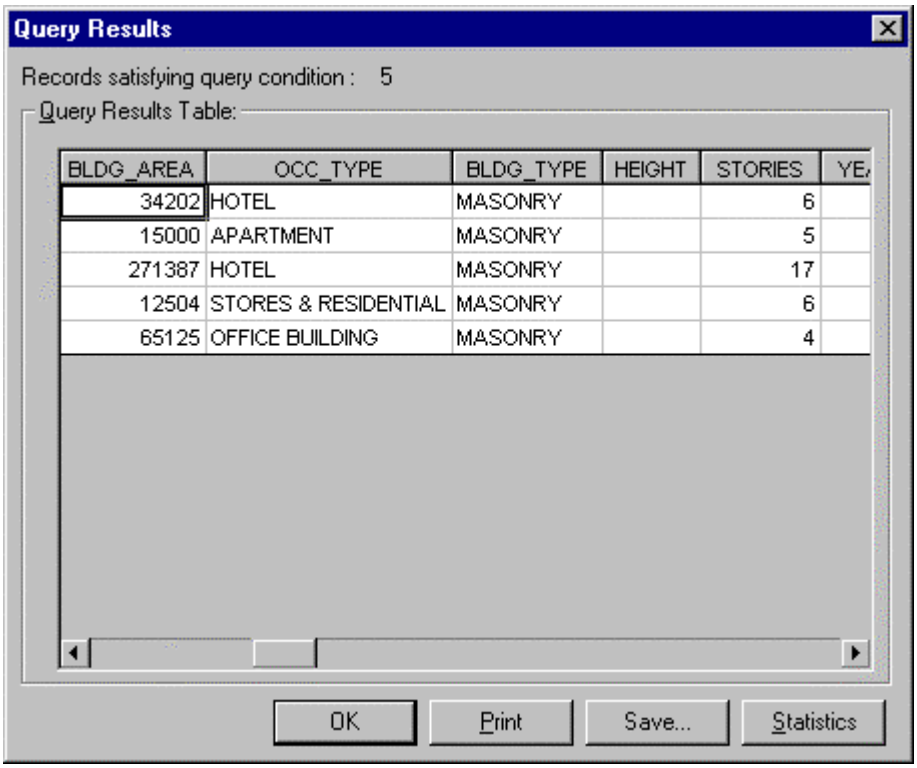


Figure 8.42 Results of query

8.8.1 Errors with the Query Tool

If you create a query statement that cannot be interpreted by the query tool you will get an error message such as the one in Figure 8.44. Here the query tool has tried to find properties where the ID_NUMBER column contains the letter C, as shown in Figure 8.43. Since the ID Number is not a letter the query tool has returned an error.

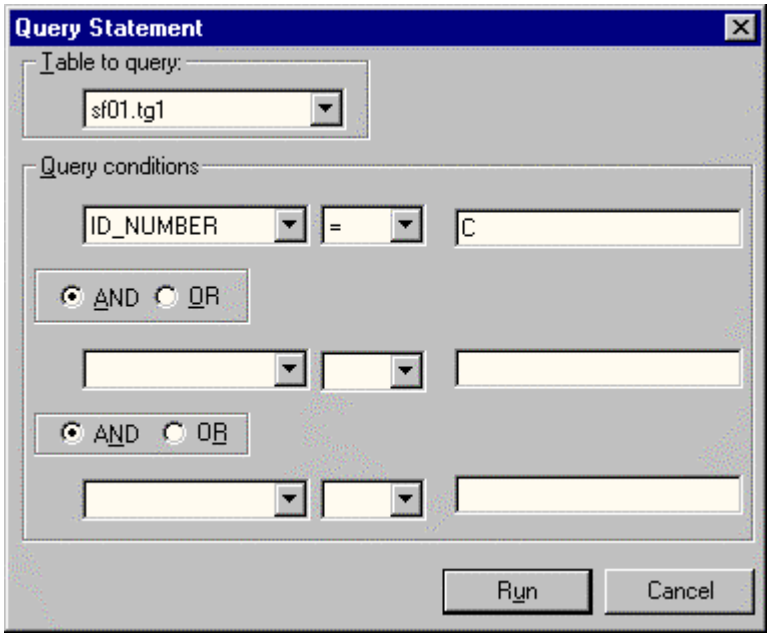


Figure 8.43 An incorrect query

Figure 8.44 Error message for an incorrect query

Similarly, the query has tried to obtain statistics for a non-numeric field and has generated the error message in Figure 8.45.

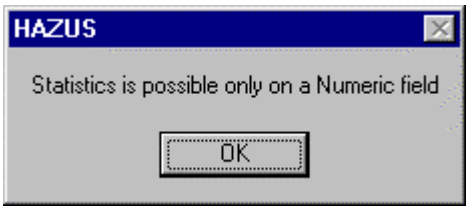


Figure 8.45 Error message for statistics tool

8.9 Exporting a Database

The databases that are created by the **BIT** tool can be quite large. It is unlikely that they will fit on a floppy disk unless they are compressed. The Export utility shown in Figure 8.46 compresses a database for transfer onto a floppy disk. Simply click on the Export button as shown in Figure 8.46 to use the Export utility.

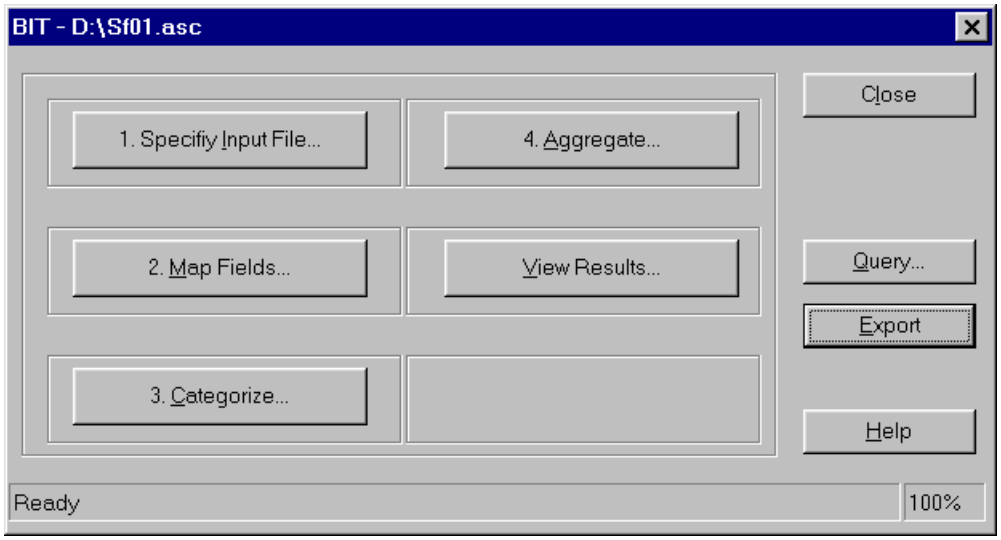


Figure 8.46 Export utility

After clicking on the Export button, the Save As dialog appears as shown in Figure 8.47. Click the Save button to complete compressing the database.

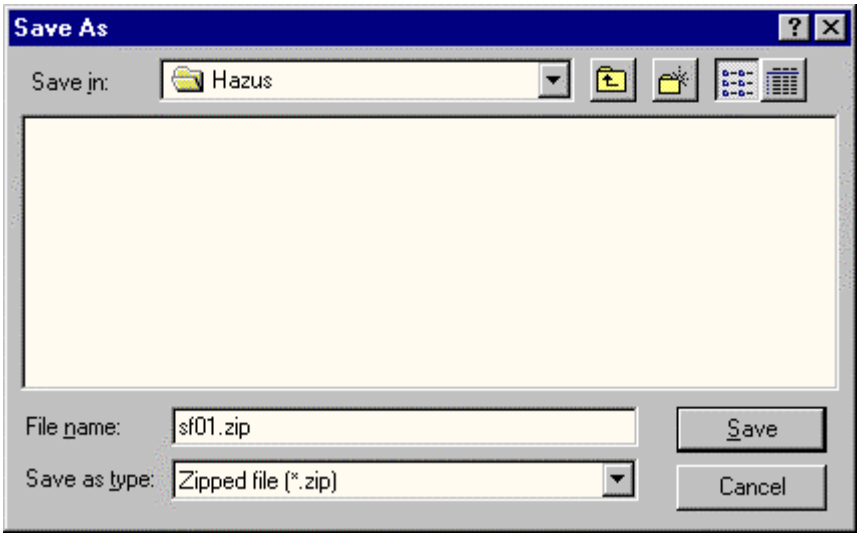


Figure 8.47 Compress and saving the database.